

Agility from First Principles: Reconstructing the Concept of Agility in Information Systems Development

Kieran Conboy

National University of Ireland, Galway, Ireland, kieran.conboy@nuigalway.ie

Awareness and use of agile methods has grown rapidly among the information systems development (ISD) community in recent years. Like most previous methods, the development and promotion of these methods have been almost entirely driven by practitioners and consultants, with little participation from the research community during the early stages of evolution. While these methods are now the focus of more and more research efforts, most studies are still based on XP, Scrum, and other industry-driven foundations, with little or no conceptual studies of ISD agility in existence. As a result, this study proposes that there are a number of significant conceptual shortcomings with agile methods and the associated literature in its current state, including a lack of clarity, theoretical glue, parsimony, limited applicability, and naivety regarding the evolution of the concept of agility in fields outside systems development. Furthermore, this has significant implications for practitioners, rendering agile method comparison and many other activities very difficult, especially in instances such as distributed development and large teams that are not conducive to many of the commercial agile methods. This study develops a definition and formative taxonomy of agility in an ISD context, based on a structured literature review of agility across a number of disciplines, including manufacturing and management where the concept originated, matured, and has been applied and tested thoroughly over time. The application of the taxonomy in practice is then demonstrated through a series of thought trials conducted in a large multinational organization. The intention is that the definition and taxonomy can then be used as a starting point to study ISD method agility regardless of whether the method is XP or Scrum, agile or traditional, complete or fragmented, out-of-the-box or in-house, used as is or tailored to suit the project context.

Key words: agile; systems development; conceptual research; agile manufacturing; agile management

History: Sandra Slaughter, Senior Editor and associate Editor. This paper was received on June 1, 2007, and was with the authors 9½ months for 3 revisions. Published online in *Articles in Advance* August 31, 2009.

The Emergence of Agile Methods

The last 10 years or so have seen the emergence of a number of information systems development (ISD) methods,¹ which have collectively been labeled as *agile*. Some of the most popular include

¹The term *ISD method* is often vaguely defined, and extensive research has attempted to provide some clarity, distinguishing *ISD method* from *methodology* (e.g., Avison and Fitzgerald 2003, Checkland 1981, Oliga 1988, Vonk 1990, Welke 1983, Brinkkemper 1996), *project management method* (e.g., Brinkkemper 1990), *process* (e.g., Connors 1992, Fitzgerald et al. 2002), and *practices* (e.g., Iivari et al. 2000, Wynekoop and Russo 1995). The interpretation of method in this study is an inclusive one, amalgamating the definitions of Hirschheim et al. (1995), Brinkkemper (1990), and Wynekoop and Russo (1995):

An ISD method encompasses the complete range of practices involved in the process of designing, building, implementing, and maintaining an information system, how these activities are accomplished and managed, the sequence and frequency of these activities, as well as the values and goals of all of the above.

eXtreme Programming (XP) (Beck 1999), the *Dynamic Systems Development Method (DSDM)* (Stapleton 1997), *Scrum* (Schwaber and Beedle 2002), *Crystal* (Cockburn 2001b), *Agile Modeling* (Ambler 2002), *Feature Driven Design (FDD)* (Coad et al. 1999), and *Lean Software Development (LSD)* (Poppendieck 2001), along with variants of each, e.g., XP-Lite (Aveling 2004).² These

²Methods are often distributed and communicated in different ways, namely through manuals, research papers, consulting, mentoring, etc. For consistency this study will refer to the version of each method in the associated reference above. It is important to distinguish between the “documented” method and the way it may be used in a particular situation (commonly referred to as “method-in-action” (Fitzgerald et al. 2002)). Iivari and Maansaari (1998) suggest that a method may be interpreted not as a set of “rules,” but as “an ideal in the sense that it is not expected to be followed literally.” In more recent years the notion of method as an idea and enactment, as opposed to a rigorous prescription, has become widely accepted in ISD thinking and is therefore the interpretation adopted in this study.

methods have been well received by those in ISD, and there is strong anecdotal evidence to suggest that awareness and indeed use of these methods is highly prevalent across the community.

Problem Statement

As is often the case with new and emerging phenomena in ISD, agile method practice has led research, with the creation, promotion, and dissemination of these methods almost completely due to the efforts of practitioners and consultants. Now agile method research is beginning to gain momentum, as is evident from the increasing number of dedicated journal special issues, conferences, conference tracks, and workshops. As a result of the dominating role industry played in the early years, however, almost all existing agile method research is now built on these industry-driven methods. Although many different studies have attempted to answer many different questions, the overwhelming majority base their study on XP, Scrum, or an individual practice within these methods. In many ways, this is very positive and represents a welcome departure from much IS research, which is all too often maligned for being pretentious and lacking relevance to practice (Benbasat and Zmud 1999, Galliers 1994).

While there are merits to adopting such a practice-oriented focus, little if any research effort has focused on the conceptual development of agility in the ISD field. Consequently, it can be argued that the current body of agile method knowledge now suffers from a number of conceptual problems as follows.

Lack of Clarity. One of the most fundamental attributes of a concept is that it is clearly communicated and understandable (Metcalf 2004; Dubin 1976, 1978; Weick 1989). While early promotional material such as the agile manifesto (Fowler and Highsmith 2001) portrays agile ISD as a clear, simple, and cohesive concept, the reality is much different. Firstly, hardly any two agile method texts or papers adopt the same definition of *agility* or *agile method*. Agility as a concept is highly multifaceted and has been used by many different people to refer to very different phenomena. To state that a particular method is or is not agile is almost meaningless given the lack of consensus on what the term “agile” refers to. Secondly, many different agile methods, variants, and

derivatives exist, and yet it is not so much the number of methods that causes the problem but the fact that these are so disparate. Some represent prescriptive operational instructions for developers (i.e., XP), others bear closer resemblance to project management methods rather than ISD methods per se (i.e., Scrum), and yet more can best be described as a set of philosophical principles (i.e., Poppendieck’s Lean Software Development). In extreme cases these methods are even contradictory (e.g., XP requires collective code ownership, FDD demands individual code ownership). It is logical and perhaps inevitable that different method creators will have different ideas on how agility can be achieved, and providing a diverse range of options for ISD teams must be beneficial for them. However, it could prove very challenging and confusing for ISD teams who wish to be agile when they are given completely conflicting, polar opposite advice. Such inconsistency and contradiction evokes similar emotive descriptions to those leveled at IS research in general, such as “fragmented adhocracy” (Banville and Landry 1989), and “crucial but confused” (Checkland and Holwell 1998).

Lack of “Theoretical Glue.” Behind any good concept or theory there should be a strong underlying logic and rationale. Whetten (1989) refers to this as a “theoretical glue” that should bind all the factors together. However, an analysis of the practices within XP, Scrum, and other commercially labeled agile methods reveals that while some have clear links to the concept of agility in a traditional sense, the connecting logic behind many others is not as clear. System metaphor and pair programming again come to mind; while short iterations, close links with the customer and co-location have related concepts in agile manufacturing and agile management; where in agile manufacturing or management have we ever heard of agility being increased by people working in pairs?

Lack of Cumulative Tradition. A good concept or theory should cumulatively build on existing research (Dubin 1978), yet this is something that IS researchers have not done particularly well (Benbasat and Zmud 1999; Keen 1991, 1980). Keen (1991) notes that most concepts and areas of concern in IS research are not as “new” as often claimed, and often “turn out to have long roots.” This trend seems to continue where agile method research is concerned. Firstly,

some (such as Boehm and Turner 2003) have noted that agile methods are incorrectly perceived as revolutionary and that there is a false dichotomic split between agile and every other method that went before, where commentators fail to realize that most agile practices have origins in much older methods. Secondly, while many researchers are studying agile methods and learning many valuable lessons, there are few that compare agile method projects with those using traditional approaches, which one would expect when a new range of methods emerges and claims to be superior. Thirdly, one would expect that studies of ISD agility would have drawn on the existing bodies of knowledge regarding agility, flexibility, and leanness in other disciplines such as management and manufacturing, where these concepts originated, matured, and have been applied and tested thoroughly over time. While there are occasional references, there are hardly any ISD studies that embrace and reflect on this tradition, with most empirical studies of agile methods using XP, Scrum, or some other commercial agile method or individual practice as a point of departure from the literature. Furthermore, there is evidence to suggest that this lack of cumulative reflection on past research is continuing. For example, in the new body of agile method research, there are many studies documenting how agile methods have been tailored, fragmented, and even combined (e.g., Fitzgerald et al. 2006, Aydin et al. 2004, Bowers et al. 2002, Cao et al. 2004, Fenwick 2003). Yet rather than taking the new method variant forward, subsequent researchers inevitably return to “base camp” and relay yet another account of how the original textbook version was tailored or customized. While such studies still have merit, it might be beneficial to see a variant from one study taken forward by multiple subsequent researchers and subjected to further implementation, stress-testing, validation, and refinement. Rather than having countless variants of a commercial method, a survival-of-the-fittest approach such as this might then see the emergence of a new, well-tested, robust method variant.

Lack of Parsimony. Authoritative works on concept development usually advocate a parsimonious approach, removing any factors that provide little additional value to our understanding (Whetten 1989). However, if we were to compile a list of all

ISD methods, practices, processes, tools, and artifacts that are commercially labeled as agile, then we would surely find redundancy and duplication. Even within individual agile methods there is a lot of nonvalue-adding content. For example, XP’s system metaphor practice is rarely used despite the popularity of the parent method (Succi and Marchesi 2001, Khaled et al. 2004, Fowler 2001). If so, then one would logically expect that method developers either provide more substantive advice on how the barriers affecting adoption of that practice can be overcome, or else issue a revised version of the method description with the redundant practice removed. Even the most recent studies involving XP still include this practice. A cohesive research effort would continuously reflect on such anomalies and avoid wasting time on them in future.

Limited Applicability. The range of applications of a concept is a key criterion for judging its quality (Metcalf 2004; Dubin 1976, 1978; Weick 1989), so agile methods should be applicable in a wide variety of ISD contexts. Irrespective of justification on purely conceptual grounds, much research in the ISD community itself has highlighted the importance of broad ISD method applicability and have called on method developers to consider ways in which their methods can be “transferred” from concept to as wide a diaspora of commercial development environments as possible (e.g., Lings and Lundell 2004, Lundell et al. 2004). Despite this, some argue commercial agile methods are largely restricted to small, co-located development teams, noncritical system development, on-demand access to developers, and many other such constraints (e.g., Stephens and Rosenberg 2003). It must be noted that some vehemently argue that agile methods are applicable in broader contexts, but the sheer volume of research attempting to rebuild and tailor these methods to various environments (e.g., Lindvall et al. 2004, Kahkonen 2004, Bowers et al. 2002, Cao et al. 2004, Crispin and House 2003, Stotts et al. 2003, Sarker 2009, Cummings et al. 2009) suggests this might not be so simple.

Implications for Practice

One may be tempted to argue that the issues described above are not that significant. After all, it is already a well-established fact that the IS literature is eclectic

and inconsistent (Checkland and Holwell 1998, Miller 1993, Backhouse et al. 1991)—universally accepted meanings of even such basic terms as *information system*, *information system development*, *method*, and *practice* have proven elusive. In fact, almost every field and discipline is saturated with ambiguous terms and inconsistent interpretations, and this is not just symptomatic of the softer sciences, as even the history of biology, chemistry, and physics is strewn with competing and conflicting definitions, theories, and models (Bryson 2004). Secondly, it is widely accepted that IS researchers are often preoccupied with theory and rigor to such an extent that relevance to practice is often forgotten (Benbasat and Zmud 1999, Galliers 1994), and one could contend that zealously chasing an overarching definition and concept of agility is yet another illustration of researchers valuing rigour at the expense of relevance. While there is certainly merit to these arguments, there is always some level of justification that can be gleaned from the oft-quoted statement “there is nothing as practical as a good theory.” However, in this case there is more substantial justification available, because the lack of conceptual foundation regarding agility in ISD does actually present significant problems for practice. Such an absence means that assessing agility is achieved by measuring compliance to commercial agile methods such as XP and Scrum. While this might yield interesting results and insights, and has underpinned many previous agile method studies, there are a number of limitations associated with such measurement as follows.

Encouragement of Method Improvement. Much research and practitioner attention has focused on the importance of continuous reflection on methods and method use, and the encouragement of ongoing process and method improvement (e.g., Paulk et al. 1995, Hunter and Thayer 2001). The position taken in this study is that, in contrast, compliance-based assessment of agility encourages a more dogmatic adherence to a method and might discourage improvement. Some agile methods such as XP are highly prescriptive and operational. An over-arching concept-centric view of agility would allow practitioners and researchers to critically reflect on methods such as XP, and constantly find ways of extending or tailoring the method to become “more agile.” A clear

symptom of this problem is that, while many studies describe cases where agile methods have been problematic and consequently tailored, there is no consolidated body of knowledge bringing these studies together and proposing how these methods can be improved upon—the textbook version of XP, Scrum, and other agile methods are still the only point of reference for developers when evaluating and choosing agile methods.

Method Comparison. As discussed earlier, commercial agile methods differ greatly in terms of abstraction, goals, practices, tools, and many other features. Therefore, method comparison is akin to comparing apples and oranges; it is extremely difficult for developers to decide which method is superior, or which should be chosen as the “baseline” method against which others are assessed. Given the number of existing agile methods, as well as the need to consider hybrids and variants, comparison of agility across methods is surely important to developers and project managers who wish to choose between methods in a structured, intelligent manner.

Assessment of Traditional or In-House Methods. Research has shown that developers rarely adhere to commercial methods, and that the vast majority use tailored variants or in-house creations. In fact, some studies show that rigorous adherence to methods among the ISD community could be as low as 5% (e.g., Hardy et al. 1995; Fitzgerald 1997, 1998). In some instances the variant or in-house method might be similar to a commercial agile method such as XP or Scrum. However, given the complex and diverse nature of systems development, the practices of an in-house method might bear little resemblance to those of any commercial agile methods, and in such cases, assessing agility by compliance might be difficult. This does not mean, however, that the method is automatically nonagile. The in-house method might fulfill all the same goals and values as XP, Scrum, and other agile methods but might do so by radically different means.

Assessment of Environments Unsuitable for Commercial Agile Methods. Rewarding compliance to a commercial agile method might provide an undesirable incentive for developers to adopt a “square peg in a round hole approach,” forcing these methods into situations to which they are not suited. An analysis

of the literature bears this out, with the vast majority of existing agile method research focusing not on small co-located teams, as was originally intended, but rather on nonstandard implementations, i.e., large teams (Lindvall et al. 2004, Kahkonen 2004, Bowers et al. 2002, Cao et al. 2004, Crispin and House 2003), start-ups (Auer and Miller 2002), distributed development environments (Kircher et al. 2001, Stotts et al. 2003), greenfield sites (Rasmusson 2003), educational environments (Fenwick 2003, Johnson and Caristi 2003, McDowell et al. 2003, Melnik and Mauer 2003, Wainer 2003), open source development (Kircher and Levine 2001), outsourcing (Kussmaul et al. 2004), and systems maintenance (Poole and Huisman 2001).

Facilitation of a Staged Conversion to Agility. Through a compliance-based view of agile method adoption, the literature identifies two ways to gradually introduce an agile method into an organization, either through “testing the water” by starting with less significant projects (e.g., Nielsen and McMunn 2005), or through the introduction of a few practices at a time (e.g., Fenwick 2003). By moving beyond a compliance model, gradual introduction can be achieved via a third alternative, namely by adopting a mildly agile practice and then replacing it with more adventurous ones in the future. So, for example, rather than adopting XP’s pair programming practice, which has proved to be highly problematic and disruptive in some circumstances (Cockburn and Williams 2001, Williams and Kessler 2000, Stotts et al. 2003, Miller 2002), pairs of developers could start by just reviewing each others code after it is written, or just pairing on certain parts of code, thus introducing the practice on a gradual basis. While much research has focused on the importance of gradual or incremental method adoption (e.g., Lundell et al. 2004, Moody 2002), the issue has not received any attention in agile method research, and as far as this researcher is aware, no agile method lists optional practices with varying degrees of extremism. While the Crystal family (Cockburn 2001b) contains methods with varying degrees of extremism, these are intended to be used in different environments and not to gradually replace each other as stepping stones to agility.

Assessment of Adherence to Overarching Values and Goals. As defined on p. 329, a “method” contains not just a set of practices or techniques but also

the values and goals that bind those practices and techniques together. It is important to look at the method level and not just the practice level to ensure that practices across a method are not conflicting, but instead are all aligned to a common set of goals and values (Brinkkemper et al. 1999, Ågerfalk and Wisstrand 2003, Ågerfalk and Fitzgerald 2005). In contrast, however, measuring compliance to practices or techniques of a commercial method such as XP focuses at an operational level and does not consider the convergent or divergent nature of the higher level goals behind those practices. An over-arching conceptual statement of agility specifying these values would allow such an assessment to take place.

The aim of this research is to provide a start toward overcoming the issues outlined above by providing a rich definition and conceptualization of agility in an ISD context, based on a literature review of agility across a number of disciplines including manufacturing and management where the concept originated, matured, and has been applied and tested thoroughly over time.

Research Approach

A methodological review of past literature is a “crucial endeavour” (Webster and Watson 2002) for any academic research, and it is vital that this is done in a rigorous and comprehensive manner (Webster and Watson 2002, Walsham 2006, Levy 2006). The literature analysis strategy and approach is rarely given much attention, with most papers elaborating more on the empirical data collection and analysis phases of the research. Because this study is based purely on a review of existing literature, however, the approach to this is now discussed in detail.

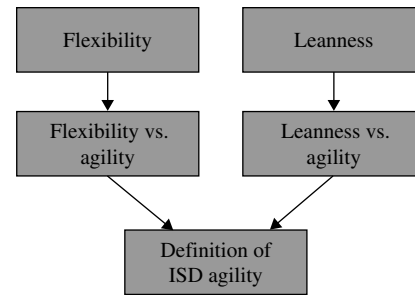
Multidisciplinary Sources of Literature and Exclusion of ISD Research. Agility is not a concept unique to ISD, first appearing in the mainstream business literature circa 1991, e.g. Goldman et al. (1991). Since then the term has become widely used across many fields and disciplines within the business domain. Research efforts within some of these fields have already focused on agility as a concept and have used conceptual and empirical means to eliminate ambiguity and to move toward a single representation of agility through the development and refinement of constructs and dimensions (e.g., Sharafi and

Zhang 1999, Vokurka and Fliedner 1998, Sharafi et al. 2001, Katayama and Bennet 1999). The purpose of this study is to contribute to ISD research by reviewing this literature and bringing it into the ISD domain. Because of this objective, and because there is a dearth of similar literature in ISD research in any case, the literature analysis strategy explicitly excluded ISD literature, and focused only on management, manufacturing, organizational behavior, and other relevant business research.

Analysis of Concepts Underpinning Agility.

While it is clear that the concept of agility has been subjected to more rigorous analysis in other disciplines and that ISD can learn from these efforts, one would be misguided to suggest that agility in other disciplines is fully and clearly understood. It would be similarly erroneous to think that all the limitations behind the way we view agility in ISD can be overcome by simply sourcing a suitable and universally accepted concept from elsewhere. Conceptual research across disciplines has shown agility to be highly polymorphous and not amenable to simple definition (Sharafi and Zhang 1999, Towill and Christopher 2002, Vokurka and Fliedner 1998, Sharafi et al. 2001, Katayama and Bennet 1999, Dove 1995). In fact, agility in manufacturing has been described as “a ragbag collection of tools and practices looking for a purpose in order to become a concept” (Sharafi et al. 2001). This suggests that the search for a definitive, all-encompassing concept of agility might not be found simply through an examination of agility in other fields; rather, it might be found through a “first-principles” approach, examining the underlying concepts of agility, namely flexibility and leanness (Sharafi and Zhang 1999, Towill and Christopher 2002) that have much earlier origins. For example, lean thinking can be traced back to the Toyota Production System in the 1950s, with its focus on the reduction and elimination of waste (Ohno 1988), the production of the Spitfire airplane in World War 2 (Childerhouse et al. 2000), and even as far back as the automotive industry in 1915 (Drucker 1995). To reflect this past, this study examines the literature on flexibility and leanness and compares and contrasts both to agility before arriving at a final conceptualized definition of agility itself.

Figure 1 Concept Development



Concept-Centric Approach. An author-centric literature review usually fails to adequately synthesize the literature and allow critical, constructive concept development and so should normally be concept-centric where possible (Levy 2006, Webster and Watson 2002). This was particularly pertinent in this study (see Figure 1), where the objective was to critically examine the concept of agility. A concept matrix was used in this study (see Figure 1), as recommended by Webster and Watson (2002) and Salipante et al. (1982), whereby the main concepts (agility, flexibility, and leanness) and their underlying subconcepts were mapped against all the literature reviewed (see excerpt in Table 1). This matrix was then used to identify the most important concepts and logical ways to group and present them. The structure for the remainder of the paper and the development of the definition and conceptualization of agility were then based on this logical grouping.

Incremental Concept Development. One potential drawback of this study is that it criticizes the current body of knowledge for having a plethora of

Table 1 Excerpt from the Concept Matrix Adopted in This Study*

Article	Concept				
	Flexibility			Leanness	
	Creation	Proaction	Reaction
Gerwin (1993)	X	X	X
De Groote (1994)		X	X
Upton (1994)		X	X
Golden and Powell (2000)		X	X
Eppink (1978)			X
Gustavsson (1984)			X
Piore (1989)	X		
.....			

*The complete matrix contained 3 concepts, 13 subconcepts against a total of 195 references.

conflicting definitions of agility, yet then paradoxically proceeds to add another one to the list. A key differentiator and contribution of this study, however, is that the definition of agility is developed incrementally, through a total of 16 iterations, unlike most studies of agility, which lead with a “take it or leave it” definition. The arguments for and against the inclusion of each conceptual component are discussed separately, along with the impact of each on the definition, thus allowing future researchers to use the parts of the definition they agree without having to take the remainder.

Consistent Abstraction. When developing any definition or concept, it is always difficult to decide what level of granularity should be used. Every researcher faces a trade-off between focus and multidimensionality and between comprehensiveness and memorability (DiMaggio 1995). In assessing the volume of literature reviewed in this study, the researcher may have been tempted to include many different concepts, philosophies, methods, tools, and practices. However, bearing in mind that the objective of this study is to present a consistent, clear definition, and conceptualization, adding some structure to agility in ISD, the researcher erred on the side of focus and memorability, adopting what Sutton and Staw (1995) call “strategic reductionism.” All the artifacts relating to agility, regardless of level of abstraction were coded and grouped into a series of high level “intellectual bins” (Miles and Huberman 1999), with the definitions and conceptual development then based on these.

Toward a Conceptualized Definition of Agility

Flexibility

Merriam-Webster’s dictionary³ highlights the complexity of the adjective *flexible*, defining it as “a ready capability for modification or change, by plasticity, pliancy, variability and often by consequent adaptability to new situations.” This complexity could be exacerbated further if one tried to understand flexibility further by drilling down on plasticity, pliancy, and each of the other terms within this definition, which by

themselves are inherently unwieldy and difficult to define. To reach a purposeful definition of ISD method flexibility, however, this research started with one of the most simple and popular interpretations of the term, namely, “the ability to adapt to change.” The first working definition for this study therefore reads simply as “the ability of an ISD method to adapt to change.”

The words “proactively” and “reactively” were then inserted. Flexibility is traditionally considered solely as a response to environmental uncertainty. Gerwin (1993), however, introduced the notion that flexibility might be used in a proactive manner, recognising the fact that an entity is not helpless while waiting for change to occur and that steps can be taken in advance of change as well as in response to it. This is supported by other conceptual studies of flexibility (De Groote 1994, Upton 1994), although this is sometimes referred to as offensive versus defensive (Golden and Powell 2000) flexibility, or initiative versus response (Goldman et al. 1995, Upton 1994). The words “proactively” and “reactively” were therefore inserted, and after making this adjustment the revised definition becomes “the ability of an ISD method to proactively or reactively adapt to change.”

The next adjustment was to insert the word “inherently” into the definition of ISD method flexibility. Robustness (Hashimoto 1980), resilience (Hashimoto et al. 1982), or mobility (Upton 1994) are included as components of flexibility, referring to the ability to endure all transitions caused by change, or the degree of change tolerated before deterioration in performance occurs without any corrective action. This concept indicates that a truly flexible entity might be capable of absorbing some change without having to take any action at all. The revised definition now appears as “the ability of an ISD method to proactively, reactively, or inherently adapt to change.”

The term “adapt to change” was then replaced with “embraces change.” When Golden and Powell (2000) made the distinction between *defensive* and *offensive* strategies discussed earlier, the rationale involved more than just the difference between taking steps before and after change. In addition, they noted that when change occurs, not only can an entity attempt to return to its original state, but it can take advantage of the change to place itself in a better position. The

³Merriam-Webster’s Collegiate Dictionary, 11th ed.

term “*adapt to*” implies that an entity is homeostatic and that its only objective in the face of change will be to return to its original state. “*Embrace*,” on the other hand, implies that the entity not only might try to return to its original state but might capitalize on the incoming change and improve its position. After making this adjustment the revised definition becomes “the ability of an ISD method to proactively, reactively, or inherently embrace change.”

The next adjustment was to insert the term “create change” into the definition. While some researchers indicate that a flexible entity can take steps to benefit from an incoming change, Gerwin (1993) and Piore (1989) extend this notion even further. As well as identifying a change, and attempting to benefit from it, flexibility can also refer to the ability “not just anticipate change, but create it” (Piore 1989), driving new change that would never have occurred were it not for the entity’s actions, e.g., creating more uncertainties for rivals, thus establishing a powerful competitive advantage (Gerwin 1993). As well as justifying the introduction of the term “create change,” this argument of Gerwin and Piore also provides further justification for the earlier replacement of “*adapt to*” with “*embrace*.” The words “*adapt to*” implies that change is the driving force and the entity’s actions are as a result of that force. “*Embrace*” signifies a two-way process where the entity not only reacts to change but can also influence it. The definition of ISD method flexibility now becomes “the ability of an ISD method to create change, or proactively, reactively, or inherently embrace it.”

The literature also highlights a distinction between flexibility that is inside and that which is outside the boundaries of the entity in question (Golden and Powell 2000). Van Oyen et al. (2001) illustrate this concept through labour flexibility, referring to internal flexibility as the ability of an organization to vary employee’s duties, working hours, or salaries; while external flexibility refers to the ability of an organization to draw resources through subcontractors, short-term contracts, or temp agencies. In the context of an ISD method as defined earlier in this study, this means that a method’s flexibility refers not just to how its practices, processes, values, and goals govern the project team but also how they facilitate and impact external people, entities, processes, and

structures. Therefore, the words “through its internal components and relationships with its external environment” are inserted to reflect the fact that an entity might not be a closed system. Rather, it might interact with other systems in its environment and might be able to use these interactions to handle change. After incorporating these changes, the revised definition of flexibility now reads as “the ability of an ISD method to create change, or proactively, reactively, or inherently embrace it, through its internal components⁴ and relationships with its environment.”

Much of the literature proposes speed of response as a key attribute of flexibility (Eppink 1978, Gustavsson 1984, Goudswaard and de Nanteuil 2000). Golden and Powell (2000) describe the temporal dimension of flexibility as the “length of time it takes for an organization to respond to environmental change” or to “adapt within a given time frame.” However, careful wording is required, as references must not include absolute speed as a measure of success. Volberda (1998) compares time taken to adapt to change against the variety of that change, acknowledging the fact that rapid response to familiar change is not necessarily better than a slightly slower response to large, strategic change. Therefore, the words “in a timely manner” are included in the definition to illustrate that it is important to consider speed of response, but that it is relative to the nature of the change being made.

Final Definition of Flexibility

The final definition of flexibility is “the ability of an ISD method to create change, or proactively, reactively, or inherently embrace change in a timely manner, through its internal components and relationships with its environment.”

Flexibility vs. Agility

In many ways, the terms *flexibility* and *agility* are very similar and have often been used interchangeably throughout the literature (Agarwal et al. 2006). An analysis of the relevant conceptual literature reveals

⁴The term “internal components” might invoke many connotations. For the purposes of this study an ISD method’s components are those incorporated into the earlier definition of an ISD method on p. 329, i.e. practices, activities, sequence or frequency of activities, values, and goals.

that the term *agile* has come to mean “proactive” (Naylor et al. 1999), “reactive” (Stratton and Warburton 2003), “change-embracing” (Goldman et al. 1995), and all the other terms that exist in the definition of flexibility proposed above.

One component of the definition that does perhaps require justification is that of change creation, as this, at least in the researcher’s mind, would not have been automatically assumed to be part of agility. However, from an analysis of the agility literature, this component clearly should be included. Agility, according to Gunasekaran and Yusuf (2002), focuses on “new ways of running businesses” and “casting off old ways of doing things.” Agility requires “distinctly aspirational tendencies” (Stratton and Warburton 2003) and involves “exploration” (Yusuf et al. 1999), “opportunity exploitation” (Christopher and Towill 2000), “acquiring new competencies, developing new product lines, and opening up new markets” (Vonderembse et al. 2006). Gunasekaran and Yusuf (2002) cite numerous other examples in the agility literature to support this notion (e.g., Kidd 1994, Gould 1997, Hong et al. 1996, James-Moore 1996). Such literature should dispel any concerns regarding the validity of incorporating change creation within an overall theory of agility.

Although there are many conceptual similarities between flexibility and agility, there are also some key differences however, and the definition of flexibility must be changed and modified to reflect these before it can be applied to agility. Firstly, Lindbergh (1990) and Sharafi and Zhang (1999) indicate that agility is made up of two components. The first is flexibility, but it shares equal prominence with the second, which is speed. Essentially, an organization must be able to “respond flexibly” and “respond speedily” (Breu et al. 2001). Terms such as “speed” (Tan 1998), “quick” (Kusiak and He 1997, De Vor and Mills 1995, Gunasekaran et al. 2002, Yusuf et al. 1999), “rapid” (Hong et al. 1996), and “fast” (Zain et al. 2002) occur in most definitions of agility. This reference to speed was discussed within the context of flexibility and was the justification behind inserting the words “in a timely manner” in the earlier definition. However, because research on the definition of agility has placed such emphasis on rapidity, it merits an adjustment to the definition before it can be applied to the term

agile. Therefore, the words “in a timely manner” are removed and replaced by the word “rapidly.” Also, the term is inserted at a more prominent point in the definition to highlight its importance. It is coupled with the word “inherently” as an extreme example of to do something rapidly is to do something inherently, i.e., where time taken equals zero. The definition now becomes “the ability of an ISD method to rapidly or inherently, create change, or to proactively or reactively embrace change through its internal components and relationships with its environment.”

Another distinction between agility and flexibility is that where agility is concerned, there exists the assumption that change is continuous, and embracing it is an ongoing activity. This assumption was laid down in the key contribution of Goldman et al. (1995), where they described agility in general terms as “a continual readiness to change.” The flexibility literature, and therefore the definition as it stands, makes no reference to continual change as opposed to a once-off change. Therefore, the definition is modified further, replacing “the ability of an entity” with “the continual readiness of an entity.” The definition of ISD method agility now becomes “the continual readiness of an ISD method to rapidly or inherently create change or to proactively or reactively embrace change through its internal components and relationships with its environment.”

Another concept emphasized heavily in the agility literature is the inclusion of knowledge and learning, and specifically, knowledge and learning from change. Jin-Hai et al. (2003) point to various literature (e.g., Prahalad and Hamel 1990, Senge 1998), which discuss how the “learning barrier” renders “an enterprise unable to keep up with environmental change.” Jin-Hai et al. (2003) suggest that various learning activities should be used within an agile enterprise, such as R&D, procession learning or “learning by doing.” Many have tried to develop a set of concepts and characteristics of an agile entity. Across these studies, characteristics proposed include “the learning employee” (Towill and Christopher 2002), “knowledge workers” (Yusuf and Adeleye 2002) “information enrichment” (Christopher and Towill 2000), “knowledge sharing” (Breu et al. 2001), the “knowledge-driven enterprise” (Yusuf et al. 1999), and the “learning organization” (Gunasekaran and Yusuf 2002). With

so many constructs evident throughout the various models of agility, the inclusion of learning in the definition is merited. Therefore the definition now becomes “the continual readiness of an ISD method to rapidly or inherently, create change, proactively or reactively embrace change, and learn from change, through its internal components and relationships with its environment.”

For some, agile means to apply the concepts of flexibility throughout different parts of the organization and not to a specific part such as manufacturing or production processes (Katayama and Bennet 1999). This has led to the coining of terms such as “agile supply chains” (Christopher 2000), “agile decision support systems” (Huang 1999), and “agile workforce” (Van Oyen et al. 2001). Therefore, it is not appropriate to state that agility is flexibility applied to different parts of the organization. However, some suggest that agility is flexibility with an “organizational orientation” (Christopher 2000), in that it is applied collectively throughout the enterprise (Goldman et al. 1995, Preiss et al. 1996). This notion would be in line with Goldman and Nagel’s (1993) “agile enterprise,” Nagel and Dove’s (1991) opinion that agility must be viewed in a “business-wide context,” and that of Gunasekaran et al. (2002), which states that agility is “not a series of practices but a fundamental management philosophy.” To reflect these, the definition is amended, changing the words “through its internal components” to “through its collective components.” The definition now appears as

Definition of Agility Part 1

... the continual readiness of an ISD method to rapidly or inherently create change, proactively or reactively embrace change, and learn from change, through its collective components and relationships with its environment.

Leanness

In comparison to the concept of flexibility, the notion of leanness is somewhat more straightforward. Most literature traces lean thinking back to the Toyota Production System in the 1950s (Ohno 1988), although as stated earlier, some have traced leanness back to World War 2 (Childerhouse et al. 2000) and even as far back as 1915 (Drucker 1995). Early lean thinking

at Toyota was initially applied to car engine manufacturing but extended to vehicle assembly in the 1960s and the wider Toyota supply chain in the 1970s. It was only at the very late stages of this evolution that formal manuals were developed and the “secrets” of this lean approach were shared with companies outside Toyota for the first time (Hines et al. 2004). In fact, these manuals were written in Japanese, and it took almost another decade before the first English literature was available (Ohno 1988, Monden 1983, Shingo 1981, Schonberger 1982). Even then, it did not make a significant impact in the mainstream literature until MIT’s five-year study of the automotive industry identified huge productivity differences between the United States and Japan, attributing this difference to a lean approach (Womack et al. 1990).

For many years, leanness has focused on the principle of economy, although this is usually referred to as cost reduction (Ohno 1988), “the elimination of waste” (Womack et al. 1990, Naylor et al. 1999, Ohno 1988), or “doing more with less” (Towill and Christopher 2002). Utilization of all resources is maximized, and no unnecessary resources are maintained (Ohno 1988). Just-in-time production, zero inventory (Womack et al. 1990), KANBAN production (Shingo 1981, Monden 1983, Ohno 1988), and maintaining minimum reasonable inventory (MRI) (Grunwald and Fortuin 1992) are all practices and approaches that illustrate the principle of economy. Therefore, the starting definition of leanness adopted in this study is: “contribution to economy.”

Leanness also incorporates many references to quality, none more so than total quality management (TQM), a philosophy central to lean thinking (Hines et al. 2004). As the concept emerged, lean manufacturing started to encompass continuous improvement and process re-engineering to increase the quality of the products being developed. A “zero defect policy” was implemented, along with constant training and promotion of employees within the lean organization (Scholtes and Hacquebord 1988, Womack et al. 1990). The focus shifted to ensuring quality was built into the product at an early stage, “upstream” as opposed to “downstream” (Swiss 1992). There was also a move from quality as determined by the organization to quality as perceived by the customer (Hines et al. 2004). Adjusting the definition of leanness in order to

Table 2 Evolution of Lean Thinking

Phases	1980s to 1990s Awareness	1990–Mid-1990s quality	Mid-1990s to 2000 quality, cost, and delivery	2000 + Value systems
Literature theme	Dissemination of shop-floor practices	Best practice movement, benchmarking leading to emulation	Value stream thinking, lean enterprise, collaboration in the supply chain	Capability at system level
Focus	JIT practices, cost	Cost, training, and promotion; TQM, process reengineering	Cost, process-based to support flow	Value and cost, tactical to strategic, integrated to supply chain
Key business processes	Manufacturing, shop-floor only	Manufacturing and materials management	Order fulfilment	Integrated processes, order fulfilment, and new business development
Industry sector	Automotive—vehicle assembly	Automotive—vehicle and component assembly	General manufacturing—often focused on repetitive manufacturing	High and low volume manufacturing, extension into service sectors
Key authors	(Shingo 1981) (Schonberger 1982) (Monden 1983) (Ohno 1988) (Mather 1988)	(Womack et al. 1990) (Hammer 1990) (Harrison 1992)	(Womack and Jones 1994) (Womack and Jones 1996) (Lamming 1993)	(Hines and Taylor 2000) (Hines et al. 2002) (Holweg and Pil 2001)

Adapted from Hines et al. (2004).

reflect the influence of quality, it now reads as “contribution to economy and quality.”

Simplicity forms a key tenet of lean thinking. In fact much of the Toyota Production Systems is based on simple practices and approaches. In terms of workflows, cells of employees work on a product to get it through in one sweeping flow (Monden 1983). Measurement of quality is straightforward, as illustrated by Shingo’s “poka-yoke” system (Shingo 1986), which uses simple statistics and mistake-proofing whereby one employee checks the work of the previous employee. Scheduling is based on simple mechanisms such as Kanban, where a card is placed in a pile of stock to trigger a reorder when a minimum level is reached (Monden 1983). Lean organizations are usually simple structures, as exemplified by the Toyota’s “focused factories” (Monden 1983, Shingo 1981). Given the fact that simplicity is a key tenet of leanness, the definition is once again adjusted and now reads as “contribution to economy, quality, and simplicity.”

As illustrated in Table 2, while cost reduction and quality were and still are fundamental concepts within lean thinking, there was a shift in focus from cost reduction, and a “shop-floor-focus” (Womack and Jones 1996) on waste, to an approach that strove to increase overall value. Until the mid-1990s, value was viewed as being equal to cost reduction, which Hines et al. (2004) describes as a “common yet critical” misconception. Instead, this new perspective

on lean thinking aimed to increase customer value through additional products and services, as well as achieving value through the traditional route of internal cost reduction (Womack and Jones 1996, Hines et al. 2004). “Value” is therefore inserted into the definition and is awarded a prominent position, illustrating the fact that value supersedes all other components “contribution to value through economy, quality, and simplicity.”

Value is a subjective concept, however. In the context of leanness, existing conceptual research suggests that value should be measured from the customer’s viewpoint, as opposed to that of the producing organization (Holweg and Pil 2001, Hines et al. 2004, Lamming 1993). As Hines et al. (2004) state, “regardless of whether an activity appeared to be wasteful from a shop-floor point of view or be costly, it is the customer that ultimately decides what constitutes *muda* (the elimination of waste), and what does not.” Therefore, the definition is modified to include “perceived customer value” reflecting this narrower interpretation.

Final Definition of Leanness. The final definition of leanness is the “contribution to perceived customer value through economy, quality, and simplicity.”

Leanness vs. Agility

Traditionally, any conceptual comparison of agile and lean would ultimately focus on the absence of value in lean thinking and the predominance of the same in the theory of agile. One of Katayama and Bennett’s

(1999) four central tenets of agility is “delivering value to the customer.” Burgess’s (2002) model of agility discusses the customer-focused “core value stream” and its underlying “project value stream,” the latter being directly relevant to this study of ISD projects. Industry specific studies of agility also contain many references to value. For example, products based on agile dynamics have moved from being standardized to being “customer-perceived value-driven” (Sharp et al. 1999). However, as can be seen from Table 2 earlier, value has recently emerged as a key component of leanness, and so has closed the gap to some extent between lean and agile. While this might no longer represent a divide between agile and lean thinking, there are still a number of other distinctions cited in the literature, and these are discussed in turn.

The first and most distinguishing factor between lean and agile is that the former cannot cope well with variability, a fundamental requirement of the latter. Maskell (1996) stated that, “to be lean is to be good at things you can control while to be agile is to be good at things you cannot.” Christopher and Towill (2000) describe leanness as “containing little fat,” whereas agile requires little or no fat in order to be “nimble.” This is illustrated by Stratton and Warburton (2003) in Figure 2, which shows that an agile entity faces volatile demand compared to the more stable demand for standard products that a lean entity faces. No change to the definition of agility is required, however, as the ability to deal with fluctuations and variations was introduced via the definition of flexibility.

Another distinction between a lean and an agile organization is that a lean one is cost efficient and productive, while an agile one learns fast, if not initially cost efficient and productive (Booth and Hammer 1995). The fact that leanness does not encourage learning would be hotly contested by some, but in

any case, no adjustment is made to the existing definition of agility because learning was already incorporated at an earlier stage in the research process.

The previous section discussed the elimination of waste and reduction of cost to be key elements of the lean enterprise. Within the realm of agility, many have done likewise, and there are many examples of cost- and time-reduction initiatives throughout the agility literature. These include the reduction of production costs (He and Kusiak 1994), handling costs (Lee 1998), relocating costs (Lee 1998), machine reconfiguration costs (Lee 1997), lost revenue due to machine relocation (Lee 1998), part differentiation costs (He and Kusiak 1994), and order completion time (Kusiak and He 1997). However, while ultimate leanness eliminates all waste, agility requires waste to be eliminated but only to the extent where its ability to respond to change is not hindered (Young et al. 2001). This does not remove the need to be economical but only lowers its priority. This is supported by numerous pieces of research. For example, Agarwal et al. (2006) conducted a review of the literature that studies the distinction between lean and agile supply chains (e.g., Naylor et al. 1999, Bruce et al. 2004, Mason-Jones et al. 2000, Olhager 2003), and concluded that while a lean supply chain aims to be a market winner in terms of cost, its agile counterpart aims only to be a market qualifier. Instead, an agile supply chain aims to be the market winner in terms of level of service provided, level of robustness, variety of product offered, and other factors that add customer value. This emphasis on value over simple cost reduction was already discussed earlier, adding further merit to a reference to value in the definition of agility. Therefore, “the maximization of value through economy, quality, and simplicity” is taken from the earlier definition of leanness and inserted into a modified definition of agility below.

Figure 2 Demand/Product Matrix for Agile and Lean Supply

Product	Demand	
	Volatile	Stable
Special	Agile	X
Standard	X	Lean

Note. Adapted from Stratton and Warburton.

Final Definition of Agility

the continual readiness of an ISD method to rapidly or inherently create change, proactively or reactively embrace change, and learn from change while contributing to perceived customer value (economy, quality, and simplicity), through its collective components and relationships with its environment.

Taxonomy of ISD Agility

The definition proposed above provides an overarching interpretation of ISD agility. This is now translated into a formative taxonomy of ISD agility below, comprising the key components of the definition, outlining the goals an ISD method or part of an ISD method must achieve if it purports to be agile. In this form, it is easier to apply in practice and to make a distinction between something that contributes to ISD agility and something that does not.

The first part of the taxonomy shows there are four primary ways in which a practice or other method component might contribute to agility. To elaborate, a method might facilitate the creation of change. If the change is created elsewhere, however, then the method might contain proactive practices that preempt that change, or some reactive measures can take place after the change event has occurred. Finally, the method might facilitate learning from change in order to improve creation, proaction, and reaction in the future. In other words, if early cycles of a project show that creativity on the part of the developers is low, risks have not been addressed, or reaction to changes have been poor or slow, then the method should contain practices or techniques to learn from these problems to facilitate improvement over time.

Note that change is central to all four aspects of agility, which is not surprising given that change was a core aspect of the definition and every iteration of its development. There is nothing revelatory in this as the motivation underpinning the very emergence

of agile methods has always been their ability to handle change (Fowler and Highsmith 2001, Cockburn 2001a, Fowler 2000, Abrahamsson et al. 2002, Beck 1999). The significance of this taxonomy, however, is that change can be interpreted in a wider sense; agile methods such as XP and Scrum are often cited for their ability to handle requirement changes, and not necessarily all the other changes that an ISD team might have to face. By adopting this broader interpretation, a method part can be considered agile even if it contributes to change other than requirements, e.g., changes in personnel, budget, contract, or third-party hardware or software.

The second part of the taxonomy requires that a method component also contributes to perceived economy, quality, or simplicity but should not perform poorly in any of the three. For example, a 400-page requirements document detailing all specifications in minute detail might increase quality and simplicity but might not be agile if perceived economy is poor due to rapid obsolescence of the document or through its lack of use.

Finally, continual readiness of the method component is also a prerequisite. For example, acceptance tests certainly contribute to agility in some circumstances; but if it takes hours to prepare tests every time they run, then their contribution to agility is unclear; the fact that automated acceptance run instantaneously at any given moment explains why they make a valid contribution to agility.

Application of the Taxonomy

While the definition and taxonomy of ISD agility was developed in a structured, rigorous, and transparent manner, this was a purely conceptual exercise, and their application in practice is necessary if they are to make a contribution to the field. Researchers often create “trivial,” impractical concepts, taxonomies, frameworks, and theories because the construction process is restricted by methodological structures that favour validation over usefulness (Lindholm 1987), a problem that seems to be particularly prevalent in the IS field (Benbasat and Zmud 1999, Keen 1991, Lee 1999). The use of what Weick (1989) refers to as “disciplined imagination” can be used to develop a research artifact or test the application of an artifact and the extent to which it achieves its purported goals. Specifically,

Figure 3 Taxonomy of ISD Agility

1. To be agile, an ISD method component* *must* contribute to one or more of the following:
 - (i) creation of change
 - (ii) proaction in advance of change
 - (iii) reaction to change
 - (iv) learning from change
2. To be agile, an ISD method component *must* contribute to one or more of the following, and *must not* detract from any:
 - (i) perceived economy
 - (ii) perceived quality
 - (iii) perceived simplicity
3. To be agile, an ISD method component *must* be continually ready i.e. minimal time and cost to prepare the component for use.

*An ISD method component refers to any distinct part of an ISD method.

“thought trials” (Weick 1989, Van de Ven 2007) can achieve this by applying “mental tests” in an applied situation to solve a particular problem.

To test the applicability of the taxonomy developed in this study, and whether it achieves its purported goals, thought trials were conducted using two completed ISD projects (TaxSys and AccountSys) in ABC Consulting,⁵ a large multinational consulting firm. These projects were chosen as “rich” and “reveatory” cases (Yin 2003). According to both teams, each had very contrasting experiences of their agile implementation, so this was seen as an ideal opportunity to apply the taxonomy and test its applicability. Eight members from each project took part and were divided into two distinct focus groups,⁶ with the objective of analysing each practice adopted on their respective project under each heading of the agility taxonomy. The groups were required to rate each practice’s contribution as *excellent*, *good*, *mediocre*, *no perceived effect*, *poor*, or *very poor* and to provide rationale or illustrative examples to support each rating. The summarized results of the exercise are contained in Appendix A, and a detailed rationale for two of the most notable practices, the on-site customer and stand-up meeting practices, are contained in Appendixes B and C, respectively. The researcher adhered to various “best practices” and principles for ensuring effective focus groups (Morgan 1997, Greenbaum 1998). As can be seen from Table 3, both projects were similar in terms of important characteristics such as team size, duration, and project type. Prior to the commencement of the projects, both teams had received the same in-house agile method training covering the basics of XP and Scrum.

Applying the Taxonomy to Test Agility of Commercially Labeled “Agile” Practices

As outlined in the problem statement, a key problem of current agile method thinking is that some practices are now commonly referred to as agile even though the connection to the concept of agility might be tenuous, and even if this link is clear it might still be too simplistic to regard the practice as agile in

⁵ “ABC Consulting,” TaxSys, and AccountSys are organization and project pseudonyms used to protect organization anonymity.

⁶ Data collected between October 2008 and December 2008.

Table 3 Profile of Project TaxSys and AccountSys

	Project TaxSys	Project AccountSys
Project duration	19 months	21 months
Team size	20	21
Team composition	1 managing partner 2 senior managers 2 team leads 11 developers 4 testers	1 managing partner 2 senior managers 4 team leads 19 developers/testers
Location	Distributed	Distributed
Development method	Scrum/XP	Scrum/XP
Type of system developed	Web-based commercial product	Web-based commercial product
Customer type	External	External
Focus group participants	2 senior managers 1 team lead 5 developers	2 senior managers 2 team leads 4 developers

every circumstance. The taxonomy proposed in this study is generic, and its application should be able to differentiate between a practice that truly contributes to agility and one that does not.

The summarized results of the trial (Appendix A) show that, from the focus group members’ experiences on their respective projects, many practices traditionally accepted as “agile” did not contribute to agility as defined by the taxonomy in this study (i.e., contributing to at least one of *creation*, *proaction*, *reaction*, and *learning*, and at least one of *perceived economy*, *quality*, and *simplicity*). In the TaxSys case, the *on-site customer*, *stand-up meetings*, *pair programming*, and *collective code ownership* were not perceived by the that focus group as having contributed to agility, while the contribution of *automated acceptance testing* and *collective code ownership* were absent in the case of AccountSys. Among the most surprising are the *on-site customer* and *stand-up meetings* on the TaxSys project, both of which are among the most commonly known practices of XP and are usually treated as being synonymous with agility. From the detailed rationale for both practices, it is clear why the TaxSys team assigned such poor ratings. In the case of the on-site customer (Appendix B), the individual performing this role was highly expensive, and the return on investment was poor; he did not partake in many of the key activities (e.g., 27 of 113 stand-up meetings and 6 of 14 retrospectives), was too “passive,” worked different shifts to the team so was often present for as little as two hours of the teams’

working day, and was also very slow to give important feedback. The stand-up meetings (Appendix C) were similarly problematic, with delayed starts, overruns, dominance of certain team members, and a lack of access to supporting documents. By surfacing these issues, the thought trial showed that the taxonomy can be used to highlight a lack of agility in practices commonly labeled as agile.

Applying the Taxonomy to Show a Practice Is Not Agile in Every Instance

Another goal of a generic taxonomy is to evaluate the agility of a practice on a case-by-case basis, sensitive to project context; what might be agile in one instance might not be agile in another. The summarized results of the thought trial show quite a few cases where the taxonomy achieved this, showing a practice to be agile on one of the projects but not the other. These were the *on-site customer*, *stand-up meetings*, *pair programming*, and *automated acceptance testing*. Again, the on-site customer and stand-up meetings provide illustrative examples. While they provided no agility on the TaxSys project, as discussed above, the AccountSys team's experience of these practices was very different. On the AccountSys project, different people substituted in and out of this role to ensure a broad level of knowledge and expertise, and they were highly involved in all activities, attending 43 of 45 stand-up meetings for example. Among many other initiatives, they organized highly creative brainstorming sessions with multiple stakeholders. Finally, in comparison to the customer representative on the TaxSys project, the role was highly economical. Likewise, stand-up meetings were shown to be very beneficial for AccountSys, positively contributing to agility in many ways but very substantially in terms of learning, economy, and quality. Such clear and convincing rationale under these ratings shows that neither practice should automatically be deemed agile in every instance.

Applying the Taxonomy to Identify New Agile Practices

The taxonomy proposed in this study should be able to identify practices that contribute to agility even though they might not be commercially labeled or publicly accepted as agile. As can be seen from the

summarized results (Appendix A), the exercise identified three practices not traditionally regarded as agile did in fact contribute to agility as defined by the taxonomy. In the TaxSys case, a *staged commitment strategy* was found to significantly contribute to agility, while in the case of AccountSys, the use of a *multisite progress dashboard* and *cultural ambassadors* contributed. As can be seen from the rationale underpinning these ratings (presented in detail in Appendix D) each received very high ratings in almost all components of ISD agility.

Conclusions and Future Research

Identification of Conceptual Issues Regarding Agility in ISD

At the outset, this paper identified a number of significant conceptual shortcomings regarding agile methods and the associated ISD literature in its current state, namely a lack of clarity, theoretical glue, parsimony, limited applicability, and naivety regarding the evolution of the concept of agility in fields outside ISD. As well as having an obvious impact on researchers, this paper highlighted significant implications these shortcomings might have for practitioners; at present it is difficult to improve agile methods, compare agile methods, assess agility of traditional, or in-house methods, assess agility of methods unsuitable for commercial agile approaches, or to facilitate a staged conversion to agility. As far as this researcher is aware, this is the first comprehensive study to explicitly identify these problems and to discuss in detail the significant implications these have for agile method practice.

Definition and Taxonomy of ISD Agility

A key contribution of this research is the development of a definition of ISD agility and a formative taxonomy. The taxonomy was also applied to two ISD projects in a large multinational consulting organization. Using thought trials, the taxonomy showed that agility of a method and its underlying components can be tested in a generic sense; practices publicly accepted to be agile might not actually contribute to agility in every instance, and conversely, practices not traditionally recognized for their agility might make a significant contribution. This notion of agile capabilities, also referred to by Vidgen and Wang

(2009) in this journal issue, represents a distinct departure from current thinking on ISD agility, where it is often crudely measured by the number of XP or Scrum practices used, regardless of their suitability to the project context or how they are implemented. In addition, the definition and taxonomy both have implications on the conceptual shortcomings of the agile method literature discussed earlier.

Lack of Clarity. The term “agile” in ISD was described earlier as so ambiguous, so multi-faceted, and defined in so many different ways that the term has lost much of its meaning. Examples of inconsistent and even contradictory interpretations were cited. To say that this study’s definition and taxonomy of agility is superior or will become universally accepted by researchers is a lofty goal. What is significant is that they were constructed in a stepped and transparent manner, whereby 16 iterations of the definition were presented, and each modification was justified by a thorough analysis of the literature pertaining to that change. Rather than presenting yet another definition of agility to be added to those in existence, this stepped and transparent approach allows researchers to analyse the justification for each part of the definition and subsequent taxonomy. This is a feature absent from the existing range of agile method studies to date, and as far as this researcher is aware, is also absent from studies of agility across any of the other disciplines reviewed as part of this study.

Lack of Theoretical Glue. As stated earlier, it is somewhat difficult to establish the theoretical rationale for labeling some practices as “agile” despite the fact that they are commonly referred to as this in practice, e.g., metaphor or pair programming. In contrast, each subcomponent of the definition and taxonomy proposed in this study is justified by an extensive discussion of why it can be classified as “agile,” drawing on arguments from previous conceptual studies of agility outside the ISD domain (e.g., Goldman et al. 1991, 1995; Preiss 1997; Dove 1994, 1995; Kidd 1994; Zhang and Sharafi 2000).

Lack of Cumulative Tradition. It was argued earlier that the current body of knowledge on ISD agility fails to link sufficiently to existing research both within the ISD literature and further afield. While the roots of agility are sporadically referenced in agile method literature, this study documents the evolution of agility,

and also the evolution of the concepts underpinning agility, namely leanness and flexibility, which have much earlier origins. While some research has looked at each of these subconcepts in isolation, they have not been examined collectively within the broader systemic context. The conceptualization developed in this study provides a macro-level view of ISD agility and its subconcepts, and while frameworks of agility exist in other disciplines (e.g., Goldman et al. 1991, 1995; Preiss 1997; Dove 1994, 1995; Kidd 1994; Zhang and Sharafi 2000), this is the first of its kind in ISD.

Lack of Parsimony. The earlier discussion revealed a lot of redundancy and duplication across agile ISD methods and related theory. During the construction of the definition in this study, specific attention was given to removing and revising its parts, and not just continually extending, thus contributing to its parsimony. Furthermore, as discussed in the research approach, the researcher strove to maintain a consistent level of abstraction, something that is lacking across and even within some existing agile methods.

Limited Applicability. It was argued earlier that commercial agile methods are largely restricted to small, co-located development teams, working on noncritical systems with on demand access to developers, among many other such constraints. In contrast, the definition and taxonomy of ISD agility proposed in this study can be applied in any ISD context. Every method can encourage creativity, proactiveness, learning, and all other parts of the definition, although these could be achieved through very different means, depending on the project environment within which the method in question will operate.

Directions for Future Research

Given the complex nature of agility and the weak theoretical and conceptual grounding in much of the existing agile method literature, this research has made a necessary first step, providing an overarching definition and formative taxonomy of agility upon which much more can be done. It is hoped that this study will inspire others to investigate further this important area, and there are many fruitful directions future researchers could take.

Firstly, researchers could develop behavioral subconstructs under the various components of the taxonomy. If we take “learning from change” as an example,

Appendix A. Summarized Focus Group Results

Project name	Practice	Perceived customer value										Continual readiness	Contributor to agility
		Creation	Proaction	Reaction	Learning	Economy	Quality	Simplicity					
TaxSys	Sprints	Excellent	Excellent	Excellent	Excellent	Good	Excellent	Good	Excellent	Good	Yes	Yes	Yes
	Prioritization	No perceived effect	No perceived effect	Excellent	No perceived effect	Good	No perceived effect	Good	No perceived effect	Good	Yes	Yes	Yes
	On-site customer	Poor	Poor	Very poor	Very poor	Very poor	Poor	Mediocre	Poor	Mediocre	Yes	No	No
	Stand-up meeting	Poor	No perceived effect	Poor	Poor	Very poor	No perceived effect	Very poor	No perceived effect	No perceived effect	Yes	No	No
	Pair programming	Mediocre	No perceived effect	No perceived effect	Good	Poor	No perceived effect	Poor	No perceived effect	No perceived effect	Yes	No	No
	Automated acceptance testing	No perceived effect	Excellent	Excellent	No perceived effect	Excellent	Excellent	Excellent	Excellent	Excellent	Yes	Yes	Yes
	Collective code ownership	Mediocre	No perceived effect	No perceived effect	No perceived effect	No perceived effect	No perceived effect	No perceived effect	No perceived effect	Poor	Yes	No	No
	Retrospectives	No perceived effect	No perceived effect	No perceived effect	Excellent	Good	Good	Good	Good	Excellent	Yes	Yes	Yes
	Staged commitment strategy	No perceived effect	Excellent	Excellent	No perceived effect	Excellent	Excellent	Excellent	No perceived effect	Excellent	Yes	Yes	Yes
	AccountSys	Sprints	No perceived effect	Excellent	Excellent	Excellent	Good	Excellent	Good	Excellent	Good	Yes	Yes
Prioritization		No perceived effect	No perceived effect	Excellent	No perceived effect	Excellent	No perceived effect	Excellent	No perceived effect	Excellent	Yes	Yes	Yes
On-site customer		Excellent	Good	Excellent	Excellent	Excellent	Excellent	Excellent	Excellent	No perceived effect	Yes	Yes	Yes
Stand-up meeting		Good	Good	Excellent	Excellent	Excellent	Excellent	Excellent	Excellent	Good	Yes	Yes	Yes
Pair programming		Excellent	No perceived effect	No perceived effect	Excellent	Good	Excellent	Good	Excellent	No perceived effect	Yes	Yes	Yes
Automated acceptance testing		No perceived effect	Excellent	Excellent	No perceived effect	Poor	Excellent	Poor	Excellent	Poor	No	No	No
Collective code ownership		No perceived effect	No perceived effect	No perceived effect	No perceived effect	No perceived effect	No perceived effect	No perceived effect	No perceived effect	Poor	Yes	No	No
Retrospectives		Good	Good	Excellent	Excellent	Good	Excellent	Good	Excellent	Good	Yes	Yes	Yes
Multisite progress dashboard		No perceived effect	Excellent	Excellent	Excellent	Excellent	Excellent	Excellent	Excellent	Excellent	Yes	Yes	Yes
Cultural ambassadors and cross-pollination		Excellent	Excellent	Excellent	Excellent	Good	Good	Good	Good	Good	Yes	Yes	Yes

Appendix B. The On-Site Customer (OSC) Practice

Project name	Perceived customer value					Continual Contributor readiness to agility			
	Creation	Proaction	Reaction	Learning	Economy		Quality	Simplicity	
TaxSys	<p>Poor</p> <p>Limited OSC knowledge of wider business needs stifled creation of new ideas.</p> <p>OSC was "highly passive."</p> <p>No OSC involvement in "spikes" (developer experimentation sessions).</p>	<p>Poor</p> <p>OSC was "highly passive," and "showed little sign of proaction" (Project Manager).</p> <p>Many changes in client needs, processes and structure not relayed by OSC to ISD team until discovered by them.</p>	<p>Very poor</p> <p>Average of 4.3 business days for OSC to give feedback on new user stories.</p>	<p>Very poor</p> <p>Limited OSC knowledge of wider business needs and issues stifled team learning.</p> <p>OSC only attended 27 of 113 stand-up meetings, and 6 of 14 retrospectives.</p> <p>OSC did not learn from many mistakes, repeating certain mistakes continuously throughout the project e.g. using wrong version of organization process documents.</p>	<p>Very poor</p> <p>OSC paid more than twice highest paid developer.</p> <p>The OSC, 1 person of 18 accounted for a large part (14%) of project budget.</p> <p>Incompatibility between OSC and developers' working hours increased real cost per hour rate. Only 2 hours overlap on many occasions.</p>	<p>Poor</p> <p>Only OSC role was user story development and validation.</p>	<p>Mediocre</p> <p>OSC did help to simplify user stories and remove duplications and inconsistencies.</p> <p>However, OSC demanded user stories to be documented in different formats different from developer or client organization, adding significant complexity and effort.</p>	<p>Yes</p>	<p>No</p>
AccountSys	<p>Excellent</p> <p>Monthly creative brainstorming sessions organized and led by OSC.</p> <p>Sessions involved multiple customer employees, across spectrum of stakeholder groups (R&D, marketing, accounts, manufacturing).</p>	<p>Good</p> <p>OSC presented to developer team on multiple occasions.</p> <p>Presentations were tailored to cater for the developers' lack of business and domain knowledge.</p>	<p>Excellent</p> <p>Real-time involvement in user story validation, with instant feedback.</p> <p>OSC continuously issued "live" reprioritized lists within 24 hours, allowing high priority items to be introduced mid-iteration.</p>	<p>Excellent</p> <p>Broad customer knowledge of wider business needs and issues.</p> <p>Attended 43 of 45 stand-up meetings.</p>	<p>Excellent</p> <p>Relatively low cost OSC.</p> <p>OSC available full-time.</p> <p>Full overlap between OSC and developers' working hours.</p>	<p>Excellent</p> <p>Different customer employees were "revolved" in and out of the OSC role during the project, matching relevant experience to the part of the system being developed.</p> <p>Quality of product increased by intermittent access to other stakeholders via OSC.</p>	<p>No perceived effect</p>	<p>Yes</p>	<p>Yes</p>

Appendix C. Stand-Up Meeting (SUM) Practice

Project name	Perceived customer value					Continual readiness	Contributor to agility		
	Creation	Proaction	Reaction	Learning	Economy			Quality	Simplicity
TaxSys	<p>Poor SUMs highly formal, often critical, and so many team members were too shy or fearful to suggest new ideas.</p>	<p>No perceived effect</p>	<p>Poor Despite raising issues, potential problems, and required changes, many team members felt that on many occasions no responsive action was taken. Certain team member updates were often vague and of limited value in operationally reacting to change.</p>	<p>Poor Standard format, where each team member stated what went well, what did not go well, and what they were going to do next facilitated basic learning across the team. However, most team members said they paid little or no attention and just “ran through the motions.” SUMs often dominated by project manager (often as much as 75% of meeting time), and bore closer resemblance to a “sermon” than exchange of knowledge across team. SUMs held away from development area, with no access to storyboards or other artifacts to refer to during meeting.</p>	<p>Very poor Time wasted due to late arrivals at SUMs. SUMs took average of approx 50 mins, ranging from 0.5 to 1.5 hours. Many SUMs dominated by over-elaborate discussion of a single issue. SUMs attended by all regardless of perceived of a lack of relevance or usefulness.</p>	<p>No perceived effect</p>	<p>No perceived effect</p>	<p>Yes</p>	<p>No</p>

Appendix C. (Cont'd.)

Project name	Perceived customer value						Continual Contributor to agility		
	Creation	Proaction	Reaction	Learning	Economy	Quality		Simplicity	Yes
AccountSys	<p>Good</p> <p>SUMs were highly informal with all encouraged to speak their mind and suggest ideas regardless of how atypical they may be.</p> <p>At every SUM, one developer had to introduce a "break-through idea," and received feedback on its usefulness and viability.</p> <p>SUMs took place at 2 P.M. each day, allowing a partner team based in the U.S. to observe proceedings and contribute new ideas.</p>	<p>Good</p> <p>A "Barriers" section was introduced in the SUM where each team member proactively listed (i) potential issues that may inhibit their work and (ii) team members who had the expertise to help.</p>	<p>Excellent</p> <p>All obstacles or problems documented by the project manager, and a list of response actions circulated to the team.</p> <p>Updates had to be specified clearly at the task level, and accompanied by the potential impact on user stories and initial estimates.</p>	<p>Excellent</p> <p>Highly interactive sessions facilitated learning. Team members often distributed excerpts of code, diagrams, or other artifacts to support their presentation.</p> <p>SUMs held in development area to allow speakers to refer to artifacts around the room.</p>	<p>Excellent</p> <p>SUMs always started on time regardless of who was on time or late.</p> <p>SUMs limited to 15 minute maximum.</p> <p>"Facilitator" appointed to keep every team member to time.</p> <p>Unresolved issues taken "offline" afterwards and only discussed be relevant team members.</p>	<p>Excellent</p> <p>Persistent problems and defects presented at SUM for group to take away and solve.</p> <p>SUM ensured any expert was aware of a problem they could assist with.</p>	<p>Good</p> <p>Each team member left the SUM with a clear idea of (i) their tasks for the day, (ii) who they would work with that day, and (iii) which team members they had to help or advise.</p>	<p>Yes</p>	<p>Yes</p>

Appendix D. Identification of New Agile Practices

Practice (project)	Description of practice	Perceived customer value					Continual Contributor to agility			
		Creation	Proaction	Reaction	Learning	Economy		Quality	Simplicity	
Staged commitment strategy (TaxSys)	Because of a delay in confirmation of project budget, the team created three sets of requirements, based on three possible budget totals (£1.3 m, £3 m, £4.7 m). These were factored into every prioritization, retrospective, and iteration.	No perceived effect	Excellent Staging proactively protected against any decision to reduce project funding.	Excellent No effort required to downsize system scope or quality in the event of funding reduction.	No perceived effect	Excellent Ensured no wasted effort on parts of the system that may be discontinued.	No perceived effect	Excellent	Yes	Yes
Multisite progress dashboard (AccountSys)	Online dashboard which represents progress of multiple teams distributed across many locations. Real-time recording of each user story, the developer(s) assigned to that story, their location(s), the status of the story, the no. of unit tests under each story, and the no. of those successfully tested or not.	No perceived effect	Excellent Transparency removed much potential uncertainty often caused by distributed development. Allowed early identification of potential issues and conflicts between teams.	Excellent All developers in all locations instantly informed of changes in user story status, underlying unit tests, etc. Ensured complete response to change, as no task “fell between” and remained accidentally unassigned.	Excellent High transparency and visibility taught each team about the other’s abilities, progress, preferences, and cultural traits. A team member could drill-down to view and learn about technical specifications, action items, code, and tests.	Excellent No financial cost or additional documentation or processes required. Seamless, real-time updates from local monitoring systems.	Excellent Large reduction in defects. No possibility for requirements to “fall between” the teams. Transparency of progress introduced a “competitive but helathy element” between teams.	Excellent Dashboard made division of work between distributed teams very simple. Dashboard provided a single point of reference simplifying communication between teams.	Yes	Yes

Appendix D. (Cont'd.)

Practice (project)	Description of practice	Perceived customer value						Continual Contributor readiness to agility	
		Creation	Proaction	Reaction	Learning	Economy	Quality		Simplicity
Cultural ambassador and cross-pollination (AccountSys)	Member of offshore team is relocated to onshore team for six weeks duration. The person performing this role was changed every six weeks to allow all offshore members to spend time with the onshore team.	Excellent Ambassador referred to potential ideas and collaborative experimentation with offshore team.	Excellent Ambassador removed much potential uncertainty often caused by distributed development. Allowed early identification of potential issues and conflicts between teams.	Excellent Ambassador assisted with conflict resolution caused by sudden, unforeseen changes e.g. problems caused by decision to shift an extra 20% of work from onshore to offshore team.	Excellent Ambassador constantly disseminated information to the onshore team members regarding the offshore team's abilities, progress, preferences, and cultural traits. Ambassador held a fortnightly seminar on the above topics.	Good Financial cost of relocating ambassador "insignificant" relative to total project cost.	Good Customers were very pleased to see and be able to communicate face-to-face with members of the offshore team they were paying for.	Good Ambassador simplified communication between teams. Ambassador explained any anomalies or issues with any of the artifacts developed by the offshore team.	Yes

one could take subconstructs from team and organizational learning literature (e.g., Huber 1996, Argyris 1999) and adapt them if needed to suit an ISD context. A similar exercise for creation, proaction, reaction, economy, quality, and simplicity would result in a more detailed, operational classification of ISD agility. The thought trials in this paper applied the taxonomy to test whether it was viable in practice and whether it achieved its purported goals. A more detailed and operational classification with subconstructs would facilitate a more prescriptive, intricate, and exhaustive exercise.

The current taxonomy examines behaviors and perceived outcomes that contribute to ISD agility. Future research could extend this by developing a set of metrics to evaluate actual performance outcomes under each component of the taxonomy. It is well known that behaviors are not always rewarded by positive outcomes, so applying outcome measures of ISD agility across methods, method variants, organizations, and projects could reveal some very interesting insights and add credence to those who claim their methods or practices are agile.

Researchers or practitioners could also provide further practical application of the taxonomy. For example, the thought trials in this study provided a snapshot of ISD agility on the TaxSys and AccountSys projects at a single point in time. Longitudinal cases could yield further insights, identifying how quickly teams can transition to agility and how effective such a transition actually is. Aside from conceptual weaknesses of agility ISD, the paper also set out practical implications arising from these issues, and these could form the basis for further research. Researchers could use the taxonomy as a starting reference point to analyse method improvement, to compare agile methods, and to assess agility in traditional or in-house methods, or in environments unsuitable for commercial agile methods. These efforts would be even easier once a set of behavioral and/or outcome subconstructs have been developed under each component of the taxonomy.

References

Abrahamsson, P., O. Salo, J. Ronkainen, J. Warsta. 2002. Agile software development methods: Review and analysis. VTT Publications 478, Technical Research Centre of Finland, Espoo, Finland.

- Agarwal, A., R. Shankar, M. Tiwari. 2006. Modeling the metrics of lean, agile and leagile supply chains: An ANP-based approach. *Eur. J. Oper. Res.* **173** 211–225.
- Ågerfalk, P., B. Fitzgerald. 2005. Methods as action knowledge: Exploring the concept of method rationale in method construction, tailoring and use. T. Halpin, J. Krogstie, K. Siau, eds. *Proc. EMMSAD'05: Tenth IFIP WG8.1 Internat. Workshop on Exploring Modeling Methods in Systems Anal. Design*, Porto, Portugal.
- Ågerfalk, P., K. Wistrand. 2003. Systems development method rationale: A conceptual framework for analysis. *The 5th Internat. Conf. Enterprise Inform. Systems (ICEIS 2003)*, Angers, France.
- Ambler, S. W. 2002. *Agile Modeling: Best Practices for the Unified Process and Extreme Programming*. John Wiley & Sons, New York.
- Argyris, C. 1999. *On Organizational Learning*. Blackwell, London.
- Auer, K., R. Miller. 2002. *Extreme Programming Applied—Playing to Win*. Addison-Wesley, Reading, MA.
- Aveling, B. 2004. XP lite considered harmful? J. Eckstein, H. Baumeister, eds. *Extreme Programming and Agile Processes in Software Engineering*. Springer, Berlin/Heidelberg.
- Avison, D., G. Fitzgerald. 2003. *Information Systems Development: Methodologies, Techniques and Tools*. McGraw-Hill, London.
- Aydin, M., F. Harmsen, K. Van Slooten, R. Stegwee. 2004. An agile informations systems development method in use. *Turkish J. Electronic Engrg.* **12** 127–138.
- Backhouse, J., J. Liebenau, F. Land. 1991. On the discipline of information systems. *J. Inform. Systems* **1** 19–27.
- Banville, C., M. Landry. 1989. Can the field of MIS be disciplined? *Comm. ACM* **32** 48–60.
- Beck, K. 1999. *Extreme Programming Explained*. Addison-Wesley, Reading, MA.
- Benbasat, I., R. Zmud. 1999. Empirical research in information systems: The practice of relevance. *MIS Quart.* **23** 3–16.
- Boehm, B., R. Turner. 2003. Using risk to balance agile and plan-driven methods. *IEEE Software* **36** 57–66.
- Booth, C., M. Hammer. 1995. Agile manufacturing concepts and opportunities in ceramics. *Ceramic Trans.* **50** 67–76.
- Bowers, J., J. May, E. Melander, M. Baarman, A. Ayoob. 2002. Tailoring XP for large mission critical software development. D. Wells, L. Williams, eds. *XP/Agile Universe*. Springer, Chicago.
- Breu, K., C. Hemingway, M. Strathern. 2001. Workforce agility: The new employee strategy for the knowledge economy. *J. Inform. Tech.* **17** 21–31.
- Brinkkemper, J. N. 1990. Formalisation of information systems modelling. Ph.D. thesis, Catholic University of Nijmegen, Nijmegen, The Netherlands.
- Brinkkemper, S. 1996. Method engineering: Engineering of information systems development methods and tools. *Inform. Software Tech.* **38** 275–280.
- Brinkkemper, S., M. Saeki, F. Harmesen. 1999. Meta-modelling based assembly techniques for situational method engineering. *Inform. Systems* **24** 209–228.
- Bruce, M., L. Daly, N. Towers. 2004. Lean or agile: A solution for supply chain management in the textiles and clothing industry. *Internat. J. Oper. Production Management* **24** 151–170.
- Bryson, B. 2004. *A Short History of Nearly Everything*. Black Swan Publishing, London.
- Burgess, T. 2002. Enhancing value stream agility. *Eur. Management J.* **20** 199–212.
- Cao, L., K. Mohan, P. Xu, B. Ramesh. 2004. How extreme does extreme programming have to be? Adapting XP practices to large-scale projects. R. Sprague, ed. *Proc. 37th Hawaii Internat. Conf. System Sci.*, IEEE Computer Society Press, Washington, DC.
- Checkland, P. 1981. *Systems Thinking, Systems Practice*. John Wiley & Sons, Chichester, UK.
- Checkland, P., S. Holwell. 1998. *Information Systems and Information Systems*. John Wiley and Sons, Chichester, UK.
- Childerhouse, P., S. Disney, D. Towill. 2000. Speeding up the progress curve towards effective supply chain management. *Internat. J. Supply Chain Management* **5** 176–186.
- Christopher, M. 2000. The agile supply chain: Competing in volatile markets. *Indust. Marketing Management* **29** 37–44.
- Christopher, M., D. Towill. 2000. Supply chain migration from lean and functional to agile and customised. *Internat. J. Supply Chain Management* **5** 206–213.
- Coad, P., J. De Luca, E. Lefebvre. 1999. *Java Modelling in Color*. Prentice Hall, Englewood Cliffs, NJ.
- Cockburn, A. 2001a. *Agile Software Development*. Addison-Wesley, Reading, MA.
- Cockburn, A. 2001b. *Crystal Clear: A Human-Powered Software Development Methodology for Small Teams*. Addison-Wesley, Reading, MA.
- Cockburn, A., L. Williams. 2001. The costs and benefits of pair programming. G. Succi, M. Marchesi, eds. *Extreme Programming Examined*. Addison-Wesley, Reading, MA.
- Connors, D. T. 1992. Software development methodologies and traditional and modern information systems. *Software Engrg. Notes* **17** 43–49.
- Crispin, L., T. House. 2003. *Testing Extreme Programming*. Pearson, Boston.
- Cummings, J., J. Espinosa, C. Pickering. 2009. Crossing spatial and temporal boundaries in globally distributed projects: A relational model of coordination delay. *Inform. Systems Res.* **20**(3) 420–439.
- De Groote, X. 1994. The flexibility of production processes: A general framework. *Management Sci.* **40** 933–945.
- De Vor, R., J. Mills. 1995. Agile manufacturing. *Amer. Soc. Mech. Engineers* **2** 977–992.
- DiMaggio, P. 1995. Comments on “What Theory is Not.” *Admin. Sci. Quart.* **40** 391–397.
- Dove, R. 1994. The meaning of life & the meaning of agile. *Production* **106** 14–15.
- Dove, R. 1995. Measuring agility: The toll of turmoil. *Production* **107** 12–14.
- Drucker, P. 1995. The information that executives truly need. *Harvard Bus. Rev.* **73** 55–63.
- Dubin, R. 1976. Theory building in applied areas. M. Dunnette, ed. *Handbook of Industrial and Organisational Psychology*. Rand McNally, Chicago.
- Dubin, R. 1978. *Theory Development*. Free Press, New York.
- Eppink, D. 1978. Managing the unforeseen: A study of flexibility. Ph.D. thesis, Free University of Amsterdam, Amsterdam, The Netherlands.
- Fenwick, J. 2003. Adapting XP to an academic environment by phasing in practices. F. Maurer, D. Wells, eds. *XP/Agile Universe*. Springer, Berlin/Heidelberg.

- Fitzgerald, B. 1997. The use of systems development methodologies in practice: A field study. *Inform. Systems J.* 7 201–212.
- Fitzgerald, B. 1998. An empirical investigation into the adoption of systems development methodologies. *Inform. Management* 34 317–328.
- Fitzgerald, B., G. Hartnett, K. Conboy. 2006. Customising agile methods to software practices. *Eur. J. Inform. Systems* 15 197–210.
- Fitzgerald, B., N. Russo, E. Stolterman. 2002. *Information Systems Development: Methods in Action*. McGraw-Hill, London.
- Fowler, M. 2000. Put your process on a diet. *Software Development* 8 32–36.
- Fowler, M. 2001. *Extreme Programming Explained*. Addison-Wesley, Reading, MA.
- Fowler, M., J. Highsmith. 2001. The agile manifesto. *Software Development* 9 28–32.
- Galliers, B. 1994. Relevance and rigour in information systems research: Some personal reflections on issues facing the information systems research community. B. Glasson, I. Hawryszkiewicz, B. Underwood, R. Weber, eds. *Proc. IFIP TC8 Open Conf. Bus. Process Re-Engineering*, Elsevier, New York.
- Gerwin, D. 1993. Manufacturing flexibility: A strategic perspective. *Management Sci.* 39 395–410.
- Golden, W., P. Powell. 2000. Towards a definition of flexibility: In search of the holy grail? *Omega* 28 373–384.
- Goldman, S., R. Nagel. 1993. Management, technology and agility: The emergence of a new era in manufacturing. *Internat. J. Tech. Management* 8 18–38.
- Goldman, S., R. Nagel, K. Preiss. 1995. *Agile Competitors and Virtual Organisations: Strategies for Enriching the Customer*. Von Nostrand Reinhold, NY.
- Goldman, S., R. Nagel, K. Preiss, R. Dove. 1991. *Iacocca Institute: 21st Century Manufacturing Enterprise Strategy: An Industry Led View*. Iacocca Institute, Bethlehem, PA.
- Goudswaard, A., M. de Nanteuil. 2000. Flexibility and working conditions: A qualitative and comparative study in seven EU member states. Technical Report (EF0007), European Foundation for Living and Working Conditions, Dublin, Ireland.
- Gould, P. 1997. What is agility? *Manufacturing Engineer* 76 28–31.
- Greenbaum, T. 1998. *The Handbook for Focus Group Research*. SAGE, London.
- Grunwald, H., L. Fortuin. 1992. Many steps towards zero inventory. *Eur. J. Oper. Res.* 59 359–369.
- Gunasekaran, A., Y. Yusuf. 2002. Agile manufacturing: A taxonomy of strategic and technological imperatives. *Internat. J. Production Res.* 40 1357–1385.
- Gunasekaran, A., E. Tirtiroglou, V. Wolstencroft. 2002. An investigation into the application of agile manufacturing in an aerospace company. *Elsevier Technovation* 22 405–415.
- Gustavsson, S. 1984. Flexibility and productivity in complex production processes. *Internat. J. Production Res.* 22 801–808.
- Hammer, M. 1990. Reengineering work: Don't automate, obliterate. *Harvard Bus. Rev.* 68 104–111.
- Hardy, C., J. Thompson, H. Edwards. 1995. The use, limitations and customization of structured systems development methods in the United Kingdom. *Inform. Software Tech.* 37 467–477.
- Harrison, A. 1992. *Just-in-Time in Perspective*. Prentice Hall, London.
- Hashimoto, T. 1980. Robustness, reliability, resilience and vulnerability criteria for planning. *Water Resources Res.* 8 11–47.
- Hashimoto, T., D. Loucks, J. Stedinger. 1982. Robustness of water resources systems. *Water Resources Res.* 18 21–26.
- He, D., A. Kusiak. 1994. Design of products and assembly systems for agility. ASME International Mechanical Engineering Congress and Exposition, New York.
- Hines, P., D. Taylor. 2000. *Going lean: A guide for implementation*. Lean Enterprise Research Centre, Cardiff Business School, Cardiff.
- Hines, P., M. Holweg, N. Rich. 2004. Learning to evolve: A review of contemporary lean thinking. *Internat. J. Oper. Production Management* 24 994–1011.
- Hines, P., R. Silvi, M. Bartolini. 2002. Demand chain management: An integrative approach in automotive retailing. *J. Oper. Management* 20 707–728.
- Hirschheim, R., H. K. Klein, K. Lyytinen. 1995. *Information Systems Development and Data Modeling: Conceptual and Philosophical Foundations*. Cambridge University Press, Cambridge, UK.
- Holweg, M., F. Pil. 2001. Successful build-to-order strategies start with the customer. *Sloan Management Rev.* 43 74–83.
- Hong, M., S. Payander, W. Gruver. 1996. Modelling and analysis of flexible fixturing systems for agile manufacturing. *Proc. IEEE Internat. Conf. Systems, Management and Cybernetics*, Beijing.
- Huang, C. 1999. An agile approach to logical network analysis in decision support systems. *Decision Support Systems* 25 53–70.
- Huber, G. 1996. Organisational learning: The contributing processes and the literatures. M. Cohen, L. Sproull, eds. *Organisational Learning*. Sage Publishers, Newbury Park, CA.
- Hunter, R., R. Thayer. 2001. *Software Process Improvement*. IEEE Computer Society Press, Piscataway, NJ.
- Iivari, J., J. Maansaari. 1998. The usage of systems development methods: Are we stuck to old practices? *Inform. Software Tech.* 40 501–510.
- Iivari, J., R. Hirschheim, H. K. Klein. 2000. A dynamic framework for classifying information systems development methodologies and approaches. *J. Management Inform. Systems* 17 179–218.
- James-Moore, S. 1996. Agility is easy but effective agile manufacturing is not. *IEE Colloquium* 179 4–10.
- Jin-Hai, L., A. Anderson, R. Harrison. 2003. The evolution of agile manufacturing. *Bus. Process Management J.* 9 170–189.
- Johnson, D., J. Caristi. 2003. eXtreme programming and the software design course. M. Marchesi, G. Succi, D. Wells, L. Williams, eds. *Extreme Programming Perspectives*. Addison Wesley, Reading, MA.
- Kahkonen, T. 2004. Agile methods for large organisations—Building communities of practice. T. Little, ed. *Proc. Agile Development Conf. 2004*, IEEE Computer Society, Salt Lake City.
- Katayama, H., D. Bennet. 1999. Agility, adaptability and leanness: A comparison of concepts and a study of practice. *Internat. J. Production Econom.* 62 43–51.
- Keen, P. 1980. MIS research: Reference disciplines and a cumulative tradition. E. Mclean, ed. *Proc. First Internat. Conf. Inform. Systems*, Philadelphia.
- Keen, P. 1991. Relevance and rigor in information systems research: Improving quality, confidence cohesion and impact. H. Nissen, H. Klein, R. Hirschheim, eds. *Information Systems Research: Contemporary Approaches & Emergent Traditions*. North-Holland, Amsterdam.
- Khaled, R., P. Barr, J. Noble, R. Biddle. 2004. System metaphor in extreme programming: A semiotic approach. *Presented at The 7th Internat. Workshop Organ. Semiotics*, Setúbal, Portugal.

- Kidd, P. 1994. *Agile Manufacturing: Forging New Frontiers*. Addison-Wesley, Reading, MA.
- Kircher, M., D. Levine. 2001. The XP of tao: eXtreme programming of large open-source frameworks. *Extreme Programming Examined*. Addison-Wesley, Reading, MA.
- Kircher, M., P. Jain, A. Corsaro, D. Levine. 2001. Distributed extreme programming. *Proc. XP2001—eXtreme Programming and Flexible Processes in Software Engrg.* Villasimius, Sardinia, Italy.
- Kusiak, A., D. He. 1997. Design for agile assembly: An operational perspective. *Internat. J. Production Res.* 35 157–178.
- Kussmaul, C., R. Jack, B. Sponsler. 2004. Outsourcing and offshoring with agility: A case study. C. Zannier, H. Erdogmus, L. Lindstrom, eds. *XP/Agile Universe*. Springer, Calgary, Alberta, Canada.
- Lamming, R. 1993. *Beyond Partnership: Strategies for Innovation and Lean Supply*. Prentice Hall, London.
- Lee, A. 1999. Rigor and relevance in MIS research: Beyond the approach of positivism alone. *MIS Quart.* 23 29–34.
- Lee, G. 1997. Reconfigurability consideration design of components and manufacturing systems. *Internat. J. Advanced Manufacturing Tech.* 13 376–386.
- Lee, G. 1998. Designs of components and manufacturing systems. *Internat. J. Advanced Manufacturing Tech.* 13 157–178.
- Levy, Y., T. J. Ellis. 2006. A systems approach to conduct an effective literature review in support of information systems research. *Informing Sci. J.* 9 181–212.
- Lindbergh, P. 1990. Strategic manufacturing management: A proactive approach. *Internat. J. Oper. Production Management* 10 94–106.
- Lindholm, C. 1987. Alternatives to validity: Some thoughts suggested by Campbell's guidelines. *Knowledge: Creation, Diffusion, Utilization* 8 509–520.
- Lindvall, M., D. Muthig, A. Dagnino, C. Wallin, M. Stupperich, D. Kiefer, J. May, T. Kahkonen. 2004. Agile software development in large organisations. *IEEE Comput.* 37 27–34.
- Lings, B., B. Lundell. 2004. On transferring a method into a usage situation. B. Kaplan, ed. *Information Systems Research: Working Group 8.2—"Relevant Theory and Informed Practice: Looking Forward from a 20 Year Perspective."* Kluwer, Boston.
- Lundell, B., B. Lings, A. Mattson, U. Arlig. 2004. Taking steps to improve working practice: A company experience of method transfer. B. Fitzgerald, E. Wynn, eds. *IT Innovation for Adaptability and Competitiveness: IFIP Working Group 8.6*. Kluwer, Boston.
- Maskell, B. 1996. *Software and the Agile Manufacturer—Computer Systems and World Class Manufacturing*. Productivity Press, Portland, Oregon.
- Mason-Jones, R., J. Naylor, D. Towill. 2000. Engineering of the leagile supply chain. *Internat. J. Agile Management Systems* 2 54–61.
- Mather, H. 1988. *Competitive Manufacturing*. Prentice Hall, Englewood Cliffs, NJ.
- McDowell, C., L. Werener, H. Bullock. 2003. The impact of pair programming on student performance, perception and persistence. *Proc. 25th Internat. Conf. Software Engrg.*, IEEE Computer Society, Washington, DC.
- Melnik, G., F. Mauer. 2003. Agile methods in learning environments: Lessons learned. F. Maurer, D. Wells, eds. *XP/Agile Universe 2003*. Springer, Berlin/Heidelberg.
- Metcalfe, M. 2004. Theory: Seeking a plain english explanation. *J. Inform. Tech. Theory Appl.* 6 13–21.
- Miles, M., A. Huberman. 1999. *Qualitative Data Analysis*. Sage, London.
- Miller, D. 1993. MIS as a discipline. *ACM SIGCSE Bull.* 2 27–32.
- Miller, R. 2002. When pairs disagree 1-2-3. D. Wells, L. Williams, eds. *XP/Agile Universe*. Springer, Berlin/Heidelberg.
- Monden, Y. 1983. *The Toyota Production System*. Productivity Press, Portland, OR.
- Moody, D. 2002. Validation of a method for representing large entity relationship models: An action research study. *Eur. Conf. Inform. Systems*, Gdansk, Poland.
- Morgan, D. 1997. *Focus Groups as Qualitative Research*. Sage, London.
- Nagel, R., R. Dove. 1991. *21st Century Manufacturing. Enterprise Strategy*. Iacocca Institute, Lehigh University, Bethlehem, PA.
- Naylor, J., M. Naim, D. Berry. 1999. Leagility: Integrating the lean and agile manufacturing paradigm in the total supply chain. *Engrg. Costs Production Econom.* 62 107–118.
- Nielsen, J., D. McMunn. 2005. The agile journey: Adopting XP in a large financial services organisation. H. Baumeister, M. Marchesi, M. Holcombe, eds. *Proc. 6th Internat. Conf. eXtreme Programming and Agile Processes (XP 2005)*, Springer, Berlin/Heidelberg.
- Ohno, T. 1988. *The Toyota Production System: Beyond Large Scale Production*. Productivity Press, Portland, OR.
- Olhager, J. 2003. Strategic positioning of the order penetration point. *Internat. J. Production Econom.* 85 319–329.
- Oliga, J. 1988. Methodological foundations of systems methodologies. *Systems Practice* 1 87–112.
- Paulk, M., C. Weber, B. Curtis, M. Chrissis. 1995. *The Capability Maturity Model: Guidelines for Improving the Software Process*. Addison-Wesley, Reading, MA.
- Piore, M. 1989. Corporate reform in American manufacturing and the challenge to economic reform. Mimeo, Department of Political Science, Massachusetts Institute of Technology, Cambridge.
- Poole, C., J. Huisman. 2001. Using extreme programming in a maintenance environment. *IEEE Software* 18 42–50.
- Poppendieck, M. 2001. Lean programming. *Software Development Magazine* 9 71–75.
- Prahalad, C., G. Hamel. 1990. The core competence of the corporation. *Harvard Bus. Rev.* 63 79–91.
- Preiss, K. 1997. A systems perspective of lean and agile manufacturing. *Agility Global Competition* 1 57–72.
- Preiss, K., S. Goldman, R. Nagel. 1996. *Cooperate to Compete: Building Agile Business Relationships*. Von Nostrand Reinhold, New York.
- Rasmusson, J. 2003. Introducing XP into greenfield projects: Lessons learned. *IEEE Comput.* 20 21–28.
- Salipante, P., W. Notz, J. Bigelow. 1982. A matrix approach to literature reviews. B. Staw, L. Cummings, eds. *Research in Organizational Behaviour*. JAI Press, Greenwich, CT.
- Sarker, S. 2009. Exploring agility in distributed information systems development teams. *Inform. Systems Res.* 20(3) 440–461.
- Scholtes, P., H. Hacquebord. 1988. Beginning the quality transformation. *Quality Progress* 21 28–33.
- Schonberger, R. 1982. *Japanese Manufacturing Techniques: Nine Hidden Lessons in Simplicity*. The Free Press, NY.

- Schwaber, K., M. Beedle. 2002. *Agile Software Development with Scrum*. Prentice Hall, Upper Saddle River, NJ.
- Senge, P. 1998. *The Fifth Discipline*. Random House, New York.
- Sharafi, H., Z. Zhang. 1999. A method for achieving agility in manufacturing organisations: An introduction. *Internat. J. Production Econom.* **62** 7–22.
- Sharafi, H., G. Colquhoun, I. Barclay, Z. Dann. 2001. Agile manufacturing: A management and operational framework. *J. Mech. Engrg.* **21** 857–869.
- Sharp, J., Z. Irani, S. Desai. 1999. Working towards agile manufacturing in the UK industry. *Internat. J. Production Econom.* **62** 155–169.
- Shingo, S. 1981. Study of the Toyota production systems. Japan Management Association, Tokyo.
- Shingo, S. 1986. *Zero Quality Control: Source Inspection and the Poka-yoke System*. Productivity Press, Portland, OR.
- Stapleton, J. 1997. *DSDM: Dynamic Systems Development Method*. Addison-Wesley, Harlow, England.
- Stephens, M., D. Rosenberg. 2003. *Extreme Programming Refactored*. Addison-Wesley Longman Publishing Co., Boston.
- Stotts, D., L. Williams, N. Nagappan, P. Baheti, D. Jen, A. Jackson. 2003. Virtual teaming: Experiments and experiences with distributed pair programming. F. Maurer, D. Wells, eds. *Extreme Programming/Agile Universe 2003*. Springer, New Orleans.
- Stratton, R., R. Warburton. 2003. The strategic integration of agile and lean supply. *Internat. J. Production Econom.* **85** 183–188.
- Succi, G., M. Marchesi. 2001. *Extreme Programming Examined*. Addison-Wesley Longman, Reading, MA.
- Sutton, R., B. Staw. 1995. What theory is not. *Admin. Sci. Quart.* **40** 371–384.
- Swiss, J. 1992. Adapting total quality management (TQM) to government. *Public Administration Rev.* **52** 356–362.
- Tan, B. 1998. Agile manufacturing and management of variability. *Internat. Trans. Oper. Res.* **5** 375–388.
- Towill, D., M. Christopher. 2002. The supply chain strategy conundrum: To be lean or agile or to be lean and agile. *Internat. J. Logist: Res. Appl.* **5** 31–44.
- Upton, D. 1994. The management of manufacturing flexibility. *California Management Rev.* **1** 72–89.
- Van de Ven, A. 2007. *Engaged Scholarship: A Guide for Organizational and Social Research*. Oxford University Press, Oxford, UK.
- Van Oyen, M., E. Gel, W. Hopp. 2001. Performance opportunity for workforce agility in collaborative and non-collaborative work systems. *IEEE Trans.* **33** 761–777.
- Vidgen, R., X. Wang. 2009. A co-evolving systems approach to the organisation of agile software development. *Inform. Systems Res.* **20**(3) 355–376.
- Vokurka, R., G. Liedner. 1998. The journey toward agility. *J. Indust. Management Data Systems* **98** 165–171.
- Volberda, H. 1998. *Building the Flexible Firm: How to Remain Competitive*. Oxford University Press, New York.
- Vonderembse, M., M. Uppal, S. Huang, J. Dismukes. 2006. Designing supply chains: Toward theory development. *Internat. J. Production Econom.* **100** 223–238.
- Vonk, R. 1990. *Prototyping: The Effective Use of CASE Technology*. Prentice Hall, London.
- Wainer, M. 2003. Adaptations for teaching software development with eXtreme programming: An experience report. F. Maurer, D. Wells, eds. *XP/Agile Universe*. Springer, New Orleans.
- Walsham, G. 2006. Doing interpretive research. *Eur. J. Inform. Systems* **15** 320–330.
- Webster, J., J. T. Watson. 2002. Analyzing the past to prepare for the future: Writing a literature review. *MIS Quart.* **26** xiii–xxiii.
- Weick, K. 1989. Theory construction as disciplined imagination. *Acad. Management Rev.* **14** 516–531.
- Welke, R. 1983. I.S./DSS: DBMS support for information system development. P. Holsapple, A. Whinston, eds. *Database Management Systems*. Reidel, Dordrecht, The Netherlands.
- Whetten, D. 1989. What constitutes a theoretical contribution? *Acad. Management Rev.* **14** 490–495.
- Williams, L., R. Kessler. 2000. The effects of “pair-pressure” and “pair-learning” on software engineering education. *Proc. Thirtieth Conf. Software Engrg. Ed. Training*, IEEE Computer Society, Austin, TX, 59–65.
- Womack, J., D. Jones. 1994. From lean production to the lean enterprise. *Harvard Bus. Rev.* **72** 93–104.
- Womack, J., D. Jones. 1996. *Lean Thinking: Banish Waste and Create Wealth for Your Corporation*. Simon and Schuster, New York.
- Womack, J., D. Jones, D. Roos. 1990. *The Machine That Changed the World*. Rawson Associates, New York.
- Wynekoop, J. L., N. L. Russo. 1995. Systems development methodologies: Unanswered questions. *J. Inform. Tech.* **10** 65–73.
- Yin, R. 2003. *Case Study Research: Design and Methods*. SAGE Publications, Thousand Oaks, CA.
- Young, K., R. Muehlhaeuser, R. Piggin, P. Rachitrangan. 2001. Agile control systems. *J. Automobile Engrg.* **215** 189–195.
- Yusuf, Y., E. Adeleye. 2002. A comparative study of lean and agile manufacturing with a related survey of current practices in the UK. *Internat. J. Production Res.* **40** 4545–4562.
- Yusuf, Y., M. Sarhadi, A. Gunasekaran. 1999. Agile manufacturing: The drivers, concepts and attributes. *Internat. J. Production Econom.* **62** 23–32.
- Zain, M., N. Kassim, E. Mokhtar. 2002. Use of I.T. and I.S. for organisational agility in Malaysian firms. *Singapore Management Rev.* **25** 25–34.
- Zhang, Z., H. Sharafi. 2000. A methodology for achieving agility in manufacturing organisations. *Internat. J. Oper. Production Management* **20** 496–512.